

### AMENDMENTS TO THE SPECIFICATION

Please amend paragraph [0037] as follows:

[0037] The block 334 passes control to the Full Platform Simulator 210 which simulates the de-virtualized instruction code(s) at the block 316 under. As stated above, the block 318 determines whether ~~there is~~ there are additional simulated instruction codes that are to be executed in the Direct Execution Environment 204. If so, control is passed to the block 304; if not, the process ends.

Please amend paragraph [0038] as follows:

[0038] Persons of ordinary skill will appreciate that Figure 3 illustrates an example implementation only. Numerous alternatives may be made. For example, while a DEX Monitor is shown separately from a platform simulator, the two may be combined together. A DEX Monitor may monitor a Direct Execution Environment for any type of event, including non virtualization events. For example, with the example of FIG. 3, an instruction like a CPUID instruction may be executed in a Direct Execution Environment as a native instruction, or it can create a virtualization event, and be simulated in a software simulator (if it is desirable to have the simulated CPU be other than the host CPU). Further still, a DEX Monitor may switch between simulated virtual machines in a format other than a round robin format (e.g., giving one simulated CPU more execution quota than the others).